

Projeto #1 - Divisão em ASM  
Entrega: 09/mai/2019

## 1 Algoritmo

O Algoritmo 1 refere-se ao algoritmo de divisão por subtração repetida adaptado para um algoritmo de divisão completo usando adições, subtrações e comparações.

Esse procedimento sempre produz **resto não negativo** ( $R \geq 0$ ). Apesar de muito simples, é exponencialmente mais lento do que outros algoritmos lentos de divisão, como o de divisão longa. É útil se o quociente é conhecido por ser pequeno.

Algoritmo 1: Divisão por subtração repetida

```
1  #define N (Numerator) as R0 (RAM[0])
2  #define D (Denominator) as R1 (RAM[1])
3  #define Q (Quotient) as R2 (RAM[2])
4  #define R (Rest) as R3 (RAM[3])
5
6  if D = 0 then
7      divisionByZero()
8  else
9      divide(N, D)
10 endif
11
12 function divide(N, D):
13     if D < 0 then
14         (Q, R) := divide(N, -D);
15         return (-Q, R)
16     endif
17     if N < 0 then
18         (Q,R) := divide(-N, D)
19         if R = 0 then
20             return (-Q, 0)
21         else
22             return (-Q - 1, D - R)
23         endif
24     endif
25     // At this point, N >= 0 and D > 0
26     return divide_unsigned(N, D)
27 endfunction
28
29 function divide_unsigned(N, D):
30     Q := 0; R := N
31     while R >= D do
32         Q := Q + 1
33         R := R - D
34     endwhile
35     return (Q, R)
36 endfunction
37
38 function divisionByZero():
39     Q := 0
40     R := 0x7FFF // or 32767, max *positive* integer number 16-bits
41     return (Q, R)
42 endfunction
```

## 2 Projeto

- Implementar o Algoritmo 1 da divisão inteira (com quociente e resto) – em grupos de até 3 integrantes (podendo ser individual) – em ASM para o computador HACK estudado nesse curso.
- Usar o Assembler implementado pelos alunos para gerar o código de máquina (arquivo `Div.hack`);
- Juntamente com os arquivos `Div.tst` e `Div.cmp` (ver Tabela 1), fornecidos pelo professor, usar o *CPU Emulator* para verificar se o código dos alunos está correto.
- Disponibilizar os arquivos na conta de *GitHub* de todos os membros da equipe.

Tabela 1: Casos de teste

Numerador RAM[0]	Denominador RAM[1]	Quociente RAM[2]	Resto RAM[3]
42	7	6	0
3	4	0	3
26	7	3	5
0	0	0	32767
1	0	0	32767
0	3	0	0
-42	7	-6	0
-7	2	-4	1
7	-2	-3	1
-7	-2	4	1